Search Results

Search Results for: [check <sentence> (directory <or> file) <AND>((DVD and file <near>system))]
Found 4 of 106,884 searched. → Rerun within the Portal

Search within Results

GO > Advanced Search > Search Help/Tips

Sort by: Title Publication Publication Date Score Binder

Results 1 - 4 of 4 **short listing**

1 Comparing two microcomputer operating systems: CP/M amd HDOS 82%
Michael A. Pechura
Communications of the ACM March 1983
Volume 26 Issue 3
Two moderate operating systems are compared, with the intent to demonstrate how each may satisfy the needs of a different set of users. The clear superiority of either systems does not emerge, but rather a difference in “design philosophy.” A method for comparing other similar systems is demonstrated.

2 Multimedia content protection by cryptography and watermarking in 80%
tamper-resistant hardware
Feng Bao
Proceedings of the 2000 ACM workshops on Multimedia November 2000

With the rapid growth of broadband network, distribution of multimedia via Internet is a must way to go. Content protection has become one of the most significant and challenging problems of this field. In this paper, we propose a general scheme that combines public key cryptography and watermarking technology together, to achieve wonderful content protection. The scheme is reliable, flexible and efficient.

3 SuSE Linux 7.0 77%
Stew Benedict
Linux Journal January 2001

4 Building self-contained websites on CD-ROM 77%
John English
ACM SIGCSE Bulletin , Proceedings of the 4th annual SIGCSE/SIGCUE ITiCSE conference on Innovation and technology in computer science education June 1999
Volume 31 Issue 3

The BURKS Project has for the past three years produced non-profit CD-ROMs of resources for students of Computer Science. Now in its third edition, BURKS is a self-contained website which incorporates a pre-installed web browser and which now spans a set of two CD-ROMs. This paper describes the techniques used to implement this product.

Results 1 - 4 of 4 **short listing**

Search Results

Search Results for: [check <sentence> DVD<AND>((DVD and file <near>system))]
Found 1 of 106,884 searched. → Rerun within the Portal

Search within Results

GO > Advanced Search > Search Help/Tips

Sort by: Title Publication Publication Date Score Binder

Results 1 - 1 of 1 short listing

1 Comparing two microcomputer operating systems: CP/M amd HDOS 82%
Michael A. Pechura
Communications of the ACM March 1983
Volume 26 Issue 3
Two moderate operating systems are compared, with the intent to demonstrate how each may satisfy the needs of a different set of users. The clear superiority of either systems does not emerge, but rather a difference in “design philosophy.” A method for comparing other similar systems is demonstrated.

Results 1 - 1 of 1 short listing

Search Results

Search Results for: [(check <or> validate) <sentence> (file <near> system)<AND>((standard<AND>((rewritable <near> (disc or disk)))))]
Found 23 of 106,884 searched. → Rerun within the Portal

Search within Results

1 A structural view of the Cedar programming environment 96%
Daniel C. Swinehart , Polle T. Zellweger , Richard J. Beach , Robert B. Hagmann
ACM Transactions on Programming Languages and Systems (TOPLAS) August 1986
Volume 8 Issue 4
This paper presents an overview of the Cedar programming environment, focusing on its overall structure—that is, the major components of Cedar and the way they are organized. Cedar supports the development of programs written in a single programming language, also called Cedar. Its primary purpose is to increase the productivity of programmers whose activities include experimental programming and the development of prototype software systems for a high-performance personal computer. T ...

2 Distributed file systems: concepts and examples 88%
Eliezer Levy , Abraham Silberschatz
ACM Computing Surveys (CSUR) December 1990
Volume 22 Issue 4
The purpose of a distributed file system (DFS) is to allow users of physically distributed computers to share data and storage resources by using a common file system. A typical configuration for a DFS is a collection of workstations and mainframes connected by a local area network (LAN). A DFS is implemented as part of the operating

system of each of the connected computers. This paper establishes a viewpoint that emphasizes the dispersed structure and decentralization of both data and con ...

3 Evolving RPC for active storage 81%
Muthian Sivathanu , Andrea C. Arpaci-Dusseau , Remzi H. Arpaci-Dusseau
ACM SIGPLAN Notices , Tenth international conference on architectural support for programming languages and operating systems on Proceedings of the 10th international conference on architectural support for programming languages and operating systems (ASPLOS-X) October 2002
Volume 37 Issue 10
We introduce Scriptable RPC (SRPC), an RPC-based framework that enables distributed system services to take advantage of active components. Technology trends point to a world where each component in a system (whether disk, network interface, or memory) has substantial computational capabilities; however, traditional methods of building distributed services are not designed to take advantage of these new architectures, mandating wholesale change of the software base to exploit more powerful hardw ...

4 The Recovery Manager of the System R Database Manager 80%
Jim Gray , Paul McJones , Mike Blasgen , Bruce Lindsay , Raymond Lorie , Tom Price , Franco Putzolu , Irving Traiger
ACM Computing Surveys (CSUR) June 1981
Volume 13 Issue 2

5 Improving the performance of log-structured file systems with adaptive methods 80%
Jeanna Neefe Matthews , Drew Roselli , Adam M. Costello , Randolph Y. Wang , Thomas E. Anderson
ACM SIGOPS Operating Systems Review , Proceedings of the sixteenth ACM symposium on Operating systems principles October 1997
Volume 31 Issue 5

6 A fast file system for UNIX 80%
Marshall K. McKusick , William N. Joy , Samuel J. Leffler , Robert S. Fabry
ACM Transactions on Computer Systems (TOCS) August 1984
Volume 2 Issue 3

7 An open operating system for a single-user machine 78%
Butler W. Lampson , Robert F. Sproull

Proceedings of the seventh symposium on Operating systems principles December 1979

The file system and modularization of a single-user operating system are described. The main points of interest are the openness of the system, which establishes no sharp boundary between itself and the user's programs, and the techniques used to make the system robust.

8 Hints for computer system design 77%

Butler W. Lampson

Proceedings of the ninth ACM symposium on Operating systems principles October 1983

Experience with the design and implementation of a number of computer systems, and study of many other systems, has led to some general hints for system design which are described here. They are illustrated by a number of examples, ranging from hardware such as the Alto and the Dorado to applications programs such as Bravo and Star.

9 The simulation of time sharing systems 68%

Norman R. Nielsen

Communications of the ACM July 1967

Volume 10 Issue 7

The development of new large scale time-sharing systems has raised a number of problems for computation center management. Not only is it necessary to develop an appropriate hardware configuration for these systems, but appropriate software adjustments must be made. Unfortunately, these systems often do not respond to changes in the manner that intuition would suggest, and there are few guides to assist in the analysis of performance characteristics. The development of a comprehensive simul ...

10 The Legion vision of a worldwide virtual computer 64%

Andrew S. Grimshaw , Wm. A. Wulf , CORPORATE The Legion Team

Communications of the ACM January 1997

Volume 40 Issue 1

11 HFS: a performance-oriented flexible file system based on building-block compositions 58%

Orran Krieger , Michael Stumm

ACM Transactions on Computer Systems (TOCS) August 1997

Volume 15 Issue 3

The Hurricane File System (HFS) is designed for (potentially large-scale) shared-memory multiprocessors. Its architecture is based on the principle that, in order to maximize performance for

applications with diverse requirements, a file system must support a wide variety of file structures, file system policies, and I/O interfaces. Files in HFS are implemented using simple building blocks composed in potentially complex ways. This approach yields great flexibility, allowing an application ...

12 Reimplementing the Cedar file system using logging and group commit 53%

R. Hagmann
ACM SIGOPS Operating Systems Review , Proceedings of the eleventh ACM Symposium on Operating systems principles November 1987
Volume 21 Issue 5

The workstation file system for the Cedar programming environment was modified to improve its robustness and performance. Previously, the file system used hardware-provided labels on disk blocks to increase robustness against hardware and software errors. The new system does not require hardware disk labels, yet is more robust than the old system. Recovery is rapid after a crash. The performance of operations on file system metadata, e.g., file creation or open, is greatly improved. < ...

13 Disk-directed I/O for MIMD multiprocessors 51%

David Kotz
ACM Transactions on Computer Systems (TOCS) February 1997
Volume 15 Issue 1

Many scientific applications that run on today's multiprocessors, such as weather forecasting and seismic analysis, are bottlenecked by their file-I/O needs. Even if the multiprocessor is configured with sufficient I/O hardware, the file system software often fails to provide the available bandwidth to the application. Although libraries and enhanced file system interfaces can make a significant improvement, we believe that fundamental changes are needed in the file server software. We prop ...

14 Design and implementation of a distributed virtual machine for networked computers 37%

Emin Gün Sirer , Robert Grimm , Arthur J. Gregory , Brian N. Bershad
ACM SIGOPS Operating Systems Review , Proceedings of the seventeenth ACM symposium on Operating systems principles December 1999
Volume 33 Issue 5

This paper describes the motivation, architecture and performance of a distributed virtual machine (DVM) for networked computers. DVMs rely on a distributed service architecture to meet the manageability, security and uniformity requirements of large, heterogeneous

clusters of networked computers. In a DVM, system services, such as verification, security enforcement, compilation and optimization, are factored out of clients and located on powerful network servers. This partitioning of system fun ...

15 Serverless network file systems 15%
T. E. Anderson , M. D. Dahlin , J. M. Neefe , D. A. Patterson , D. S. Roselli , R. Y. Wang
ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles December 1995
Volume 29 Issue 5

16 At the Forge: Writing Modules for mod_perl 11%
Reuven M. Lerner
Linux Journal January 1999

17 A coherent distributed file cache with directory write-behind 5%
Timothy Mann , Andrew Birrell , Andy Hisgen , Charles Jerian , Garret Swart
ACM Transactions on Computer Systems (TOCS) May 1994
Volume 12 Issue 2
Extensive caching is a key feature of the Echo distributed file system. Echo client machines maintain coherent caches of file and directory data and properties, with write-behind (delayed write-back) of all cached information. Echo specifies ordering constraints on this write-behind, enabling applications to store and maintain consistent data structures in the file system even when crashes or network faults prevent some writes from being completed. In this paper we describe ...

18 The UNIX time-sharing system 4%
Dennis M. Ritchie , Ken Thompson
Communications of the ACM July 1974
Volume 17 Issue 7
UNIX is a general-purpose, multi-user, interactive operating system for the Digital Equipment Corporation PDP-11/40 and 11/45 computers. It offers a number of features seldom found even in larger operating systems, including: (1) a hierarchical file system incorporating demountable volumes; (2) compatible file, device, and inter-process I/O; (3) the ability to initiate asynchronous processes; (4) system command language selectable on a per-user basis; and (5) over 100 subsystems including a ...

19 Linux Gazette: Big Brother Monitoring System 4%

Paul M. Sittler
Linux Journal January 1997

20 Separating access control policy, enforcement, and functionality in extensible systems 3%
Robert Grimm , Brian N. Bershad
ACM Transactions on Computer Systems (TOCS) February 2001
Volume 19 Issue 1

Extensible systems, such as Java or the SPIN extensible operating system, allow for units of code, or extensions, to be added to a running system in almost arbitrary fashion. Extensions closely interact through low-latency but type-safe interfaces to form a tightly integrated system. As extensions can come from arbitrary sources, not all of whom can be trusted to conform to an organization's security policy, such structuring raises the question of how security constraints are enforced in an ...

Results 1 - 20 of 23 **short listing**

Prev Page 1 2 Next Page

Your search matched **[0]** of **[921157]** documents.

You may refine your search by editing the current search expression or entering
a new one the text box. Then click search Again.

(validate or check or verify) and (file near system) and (DVD)

Search Again

OR

Use your browser's back button to return to your original search page.

Results:

No documents matched your query.

Your search matched **[0]** of **[921157]** documents.

You may refine your search by editing the current search expression or entering
a new one the text box. Then click search Again.

(validate or check or verify) and (file <near> system) and (rewritable and (disk or disc))

Search Again

OR

Use your browser's back button to return to your original search page.

Results:

No documents matched your query.

Your search matched **[0]** of **[922932]** documents.

You may refine your search by editing the current search expression or entering
a new one the text box. Then click search Again.

(validate or check or verify) and (file near system) and (rewritable and (disk or disc))

Search Again

OR

Use your browser's back button to return to your original search page.

Results:

No documents matched your query.

Welcome to IEEE Xplore
- Home
- What Can I Access?
- Log-out

Tables of Contents
- Journals & Magazines
- Conference Proceedings
- Standards

Search
- By Author
- Basic
- Advanced

Member Services
- Join IEEE
- Establish IEEE Web Account
- Access the IEEE Member Digital Library

Print Format

Your search matched **[0]** of **[922932]** documents.

You may refine your search by editing the current search expression or entering
a new one the text box. Then click search Again.

(validate or check or verify) and (file <near> system) and rewritable

Search Again

OR

Use your browser's back button to return to your original search page.

Results:

No documents matched your query.

Home | Log-out | Journals | Conference Proceedings | Standards | Search by Author | Basic Search | Advanced Search
Join IEEE | Web Account | New this week | OPAC Linking Information | Your Feedback | Technical Support | Email Alerting
No Robots Please | Release Notes | IEEE Online Publications | Help | FAQ | Terms | Back to Top

CiteSeer Find: file management rewritable (disc | Documents | Citations

Searching for **file management rewritable (disc or disk) check file system.**

Restrict to: Header Title Order by: Citations Hubs Usage Date Try: Amazon B&N Google (RI) Google (Web) CSB DBLP

No documents match Boolean query. Trying non-Boolean relevance query.

1000 documents found. Retrieving documents... **Order: relevance to query.**

MacFS: A Portable Macintosh File System Library - Dinda, Necula, Price (1998) (Correct)
MacFS: A Portable Macintosh **File System** Library Peter A. Dinda George C. Necula
reports-archive.adm.cs.cmu.edu/anon/1998/CMU-CS-98-145.ps

Implementation and Evaluation of Prefetching in the Intel.. - Meenakshi Arunachalam (1996) (Correct)
of Prefetching in the Intel Paragon Parallel **File System** Meenakshi Arunachalam Alok Choudhary Brad
between the speeds of the I/O **system** (e.g. **disks**) and compute processors in parallel **systems**
www.ece.nwu.edu/~meena/papers/ipps.ps

The Zebra Striped Network File System - Hartman, Ousterhout (1993) (Correct) (118 citations)
1 The Zebra Striped Network **File System** John H. Hartman John K. Ousterhout
www.cs.arizona.edu/people/jhh/papers/zebra_tocs.ps

Application-Controlled File Caching Policies - Cao, Felten, Li (1994) (Correct) (52 citations)
Application-Controlled **File** Caching Policies Pei Cao, Edward W. Felten, and
ftp.cs.princeton.edu/reports/1994/445.ps.Z

File System Logging Versus Clustering: A Performance.. - Seltzer, Smith.. (1995) (Correct) (28 citations)
File System Logging Versus Clustering: A Performance
www.eecs.harvard.edu/~margo/papers/../usenix.195/usenix.195.ps.gz

A Hypertext System for Integrating Heterogeneous, Autonomous.. - Noll, Scacchi (1994) (Correct) (2 citations)
In addition, there will be a shared network **file system** so all platforms can share **files**. The group
Noll and Walt Scacchi Information and Operations **Management** Dept. University of Southern California Los
is navigational rather than query-oriented, as **disc**ussed later in section 4. A request-response style
cwis.usc.edu/dept/ATRIUM/Papers/Integrating_Software_Repositories.ps

Serverless Network File Systems - Anderson, Dahlin, Neefe, Patterson.. (1995) (Correct) (86 citations)
1 Serverless Network **File Systems** Thomas E. Anderson, Michael D. Dahlin,
fmg-www.cs.ucla.edu/classes/239_1.fall97/papers/SOSP95/serverless.ps

Cspack Client-Server Routines And Utilities - Cern (Correct)
: 3 1.1.6 FATMEN -A Distributed **File** and Tape **Management System** :3
wwwinfo.cern.ch/asdoc/./psdir/cspack.ps.gz

The Scotch Parallel Storage System - Gibson, al. (1995) (Correct) (14 citations)
error-handling in RAID subsystems fourth, a **file system** extension that allows serial programs to
Second, resource decisions, notably buffer-cache **management**, can be improved by foreknowledge. Third, deep
the latter two limitations through application **disc**losure of future accesses and application
www.cs.cmu.edu/afs/cs.cmu.edu/project/pdl/ftp/SPFS/tr95-107.ps

The Tiger Shark File System - Haskin, Schmuck (1996) (Correct) (24 citations)
The Tiger Shark **File System** Roger L. Haskin Frank B. Schmuck IBM
inherent in the design. All operator-initiated **management** functions can be performed while the **system**
a **file system** and storage node, but to keep the **disc**ussion simple, we will always treat storage and
www.research.ibm.com/webvideo/shark96.ps

The Effect of Client Caching on File Server Workloads - Kevin Froese (1996) (Correct) (6 citations)
The Effect of Client Caching on **File** Server Workloads Kevin W. Froese Richard B. Bunt
has suggested that various strategies for cache **management** may not be equally suited to the circumstances
to overcome the performance impact of speed **disc**repancies between different components of a
www.cs.usask.ca/staff/kwf230/research/hicss96.ps.gz

Amoeba made compatible with Unix: the ADE approach - Sun Keuning (Correct)
commands, nor of the number and location of the **file** servers that store their **files**. To the casual
platforms. In many aspects (e.g. process **management**, signal handling, etc.Amoeba is
incompatible aspects between Amoeba and Unix are **discussed** in Section 2. Then, the basic idea of the ADE
www.cit.gu.edu.au/~scz/papers/acsc94.ps.Z

A Quantitative Analysis of Cache Policies for Scalable.. - Michael Dahlin (1994) (Correct) (34 citations)
Analysis of Cache Policies for Scalable Network **File Systems** Michael D. Dahlin, Clifford J. Mather,
ftp.cs.berkeley.edu/ucb/people/tea/xfs.ps

Operating System Support for Easy Development of Distributed.. - Kenichi Kourai (1998) (Correct)
System Support for Easy Development of Distributed **File Systems** Kenichi Kourai, Shigeru Chiba, and Takashi
When writing it back, the protection manager **checks** the data by examining various properties of the
www-masuda.is.s.u-tokyo.ac.jp/~kourai/papers/TR98-01.ps.gz

Cluster-Based File Replication in Large-Scale Distributed.. - Harjinder Sandhu (1992) (Correct) (17 citations)
Cluster-Based **File** Replication in Large-Scale Distributed **Systems**
ftp.cs.toronto.edu/pub/reports/csrg/255/frolic.ps.Z

Disk-directed I/O for an Out-of-core Computation - Kotz (1995) (Correct) (1 citation)
Report PCS-TR95-251 January 13, 1995 Abstract New **file systems** are critical to obtain good I/O
more detailed background information. Section 3 **discusses** the LU-decomposition program. In Section 4 we
at URL ftp.cs.dartmouth.edu/TR/TR95-251.ps.Z **Disk**-directed I/O for an Out-of-core Computation David
ftp.cs.dartmouth.edu/TR/TR95-251.ps.Z

Operating System Support For Easy Development Of Distributed File .. - Kourai (1998) (Correct)
System Support For Easy Development Of Distributed **File Systems** Kenichi Kouraiy Shigeru Chibaz Takashi
The roles of the protection manager are module **management**, upcall processing, and safe manipulation of
When writing it back, the protection manager **checks** the data by examining various properties of the
www-masuda.is.s.u-tokyo.ac.jp/~kourai/papers/kourai-pdcs98.ps.gz

Intelligent, Adaptive File System Policy Selection - Tara Madhyastha (1996) (Correct) (3 citations)
Intelligent, Adaptive **File System** Policy Selection Tara M. Madhyastha
www.cs.cmu.edu/~tara/hdfpaper.ps.Z

User-mode Per-process Name Spaces for the AP1000 File System - Bradley Broom (1993) (Correct)
1 User-mode Per-process Name Spaces for the AP1000 **File System** Bradley M. Broom Brad.Broom@anu.edu.au
cs.anu.edu.au/techreports/1993/TR-CS-93-08.ps.gz

First 20 documents Next 20

Try your query at: Amazon Barnes & Noble Google (RI) Google (Web) CSB DBLP